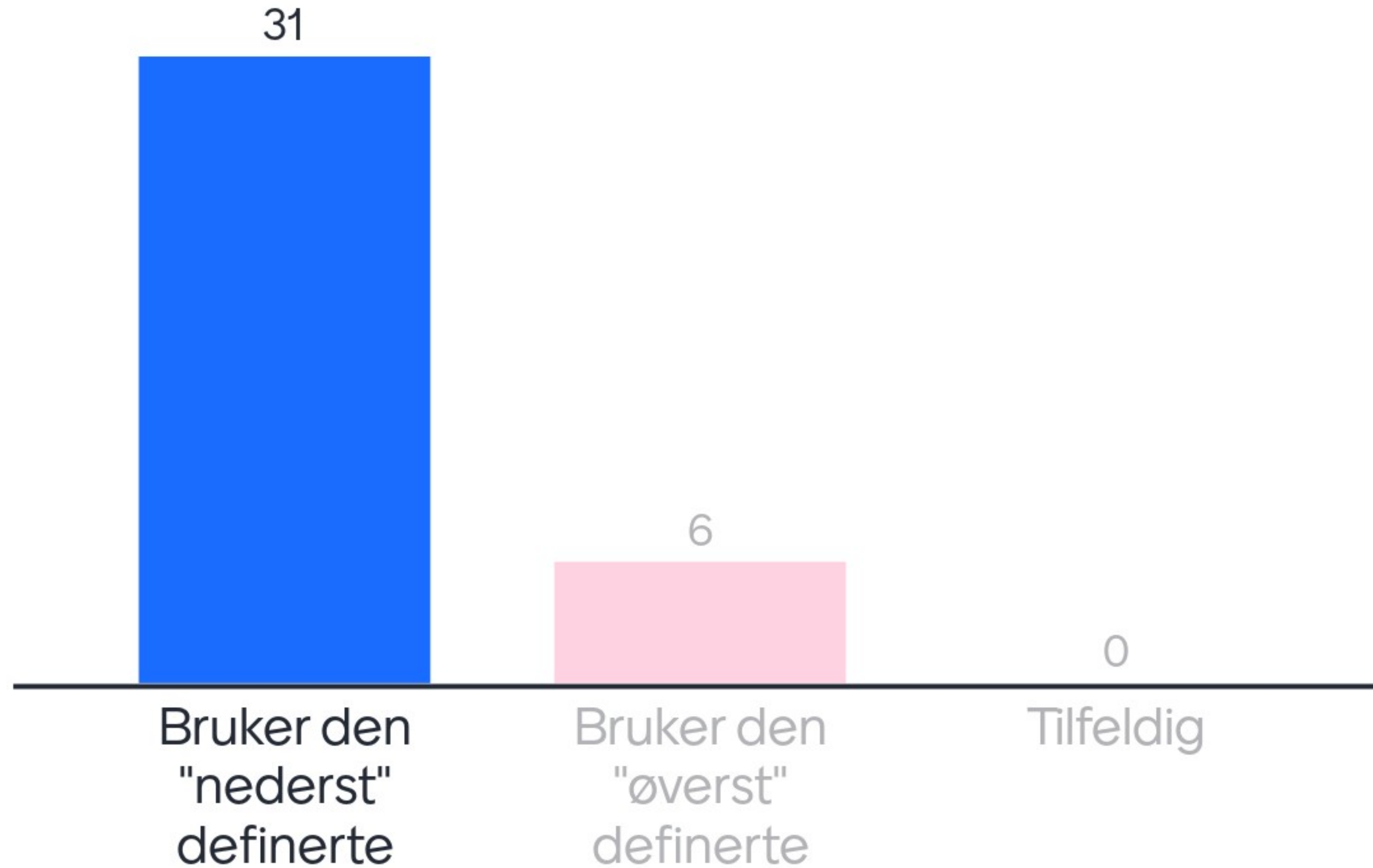


Arv, subklasser og grensesnitt

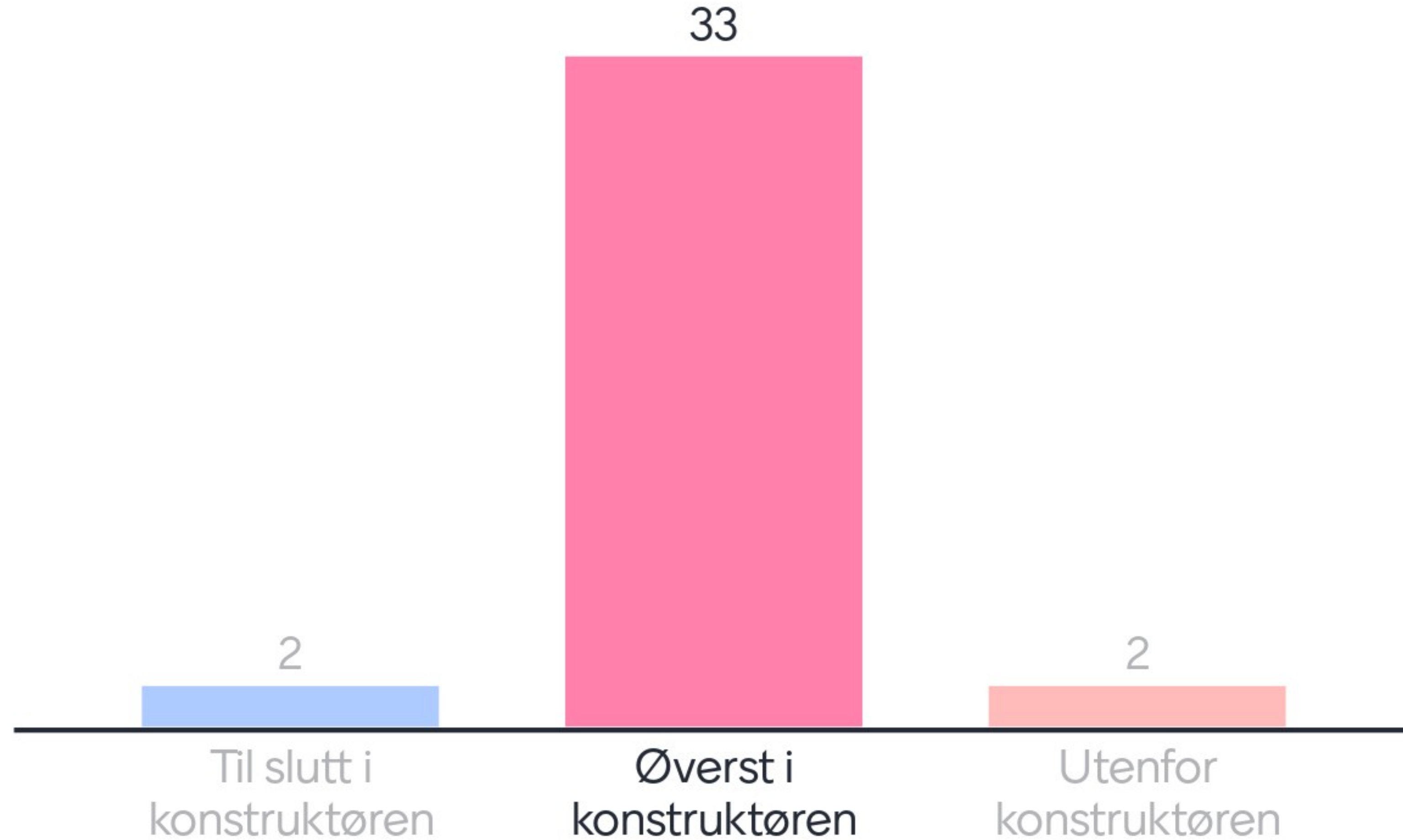
Repetisjonskurs 24.05.22



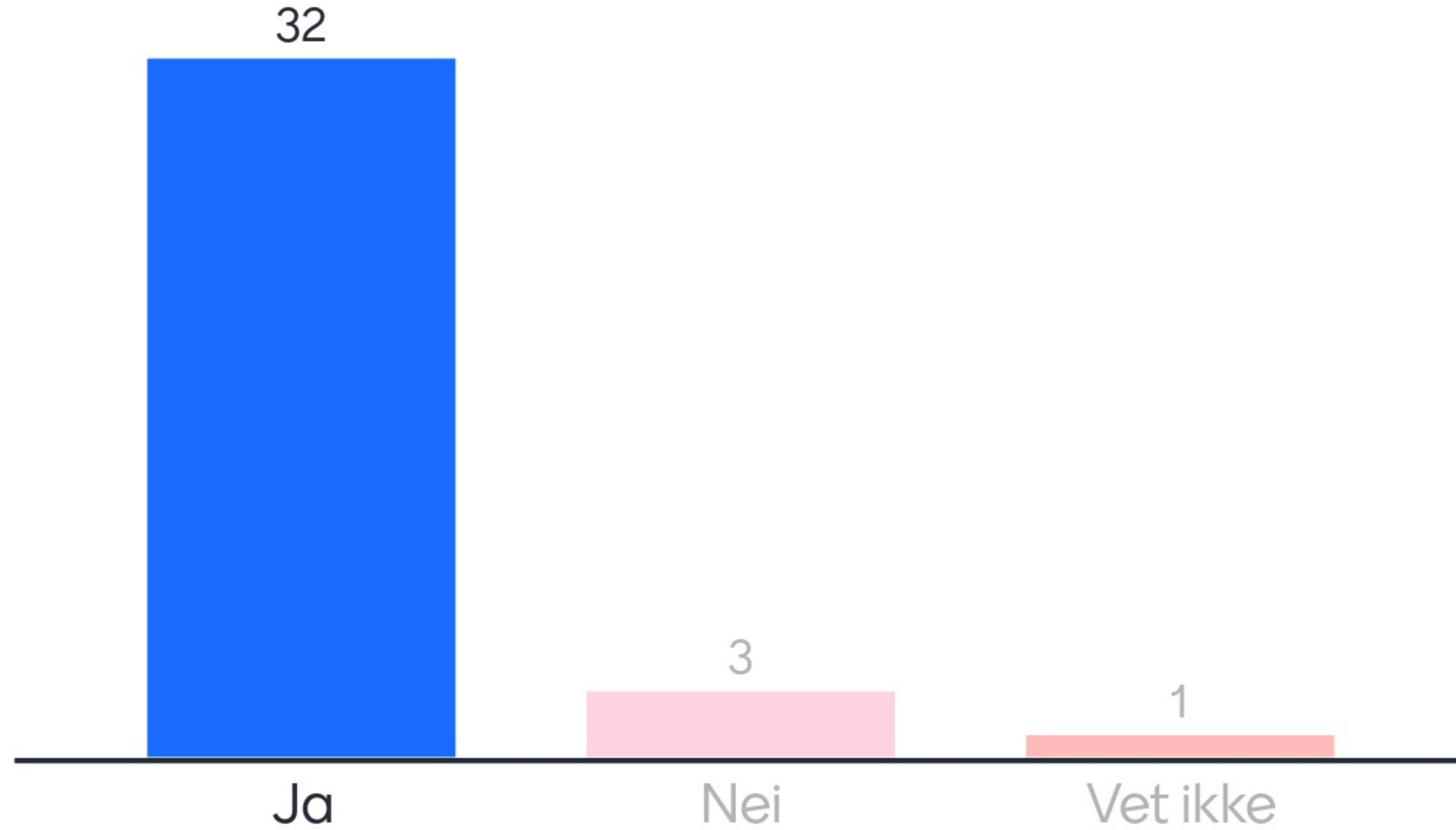
Hvordan systemet vet hvilken metode den skal benytte seg av hvis det er flere polymorfe metoder



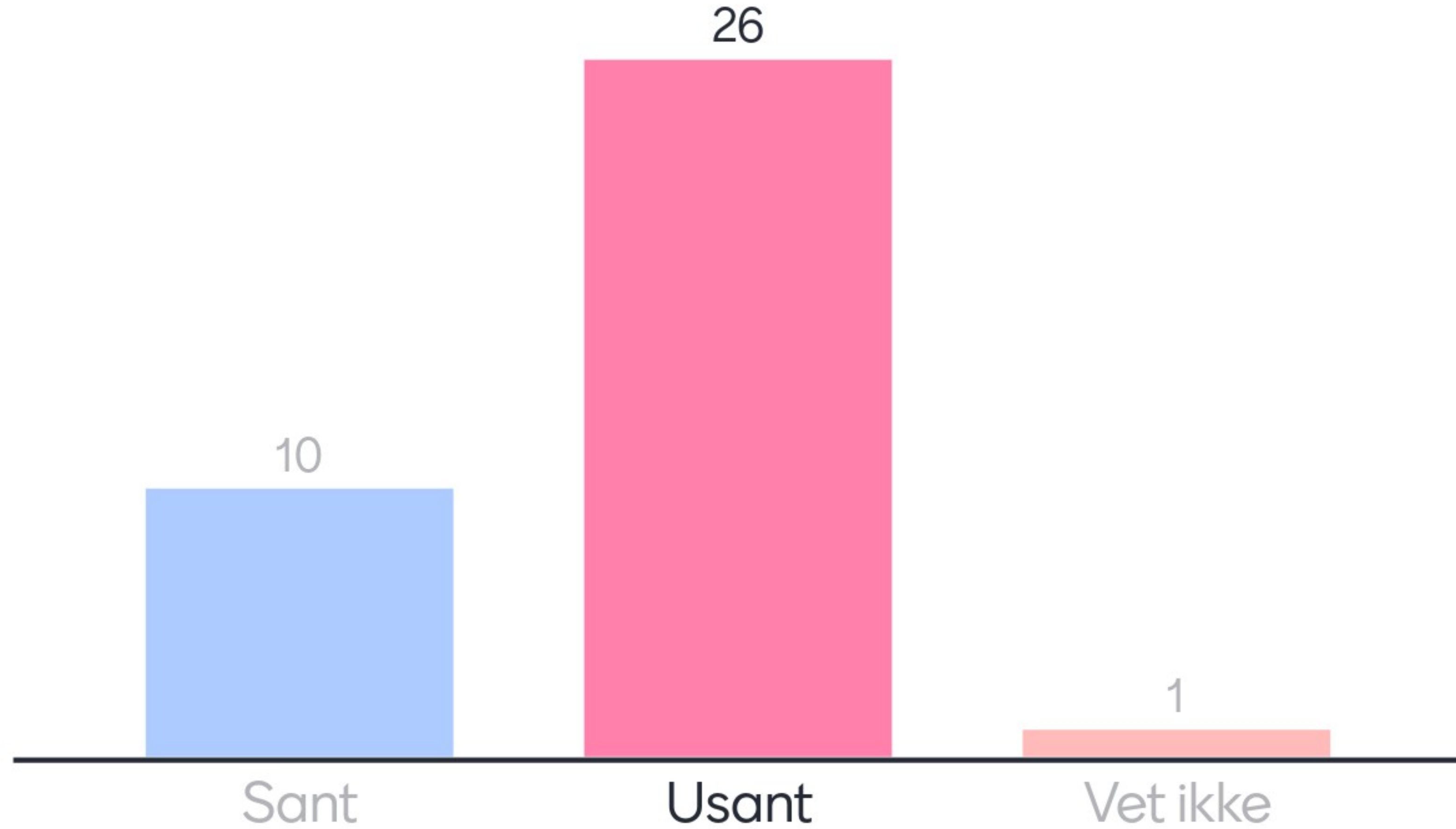
Et kall på `super()` må legges...



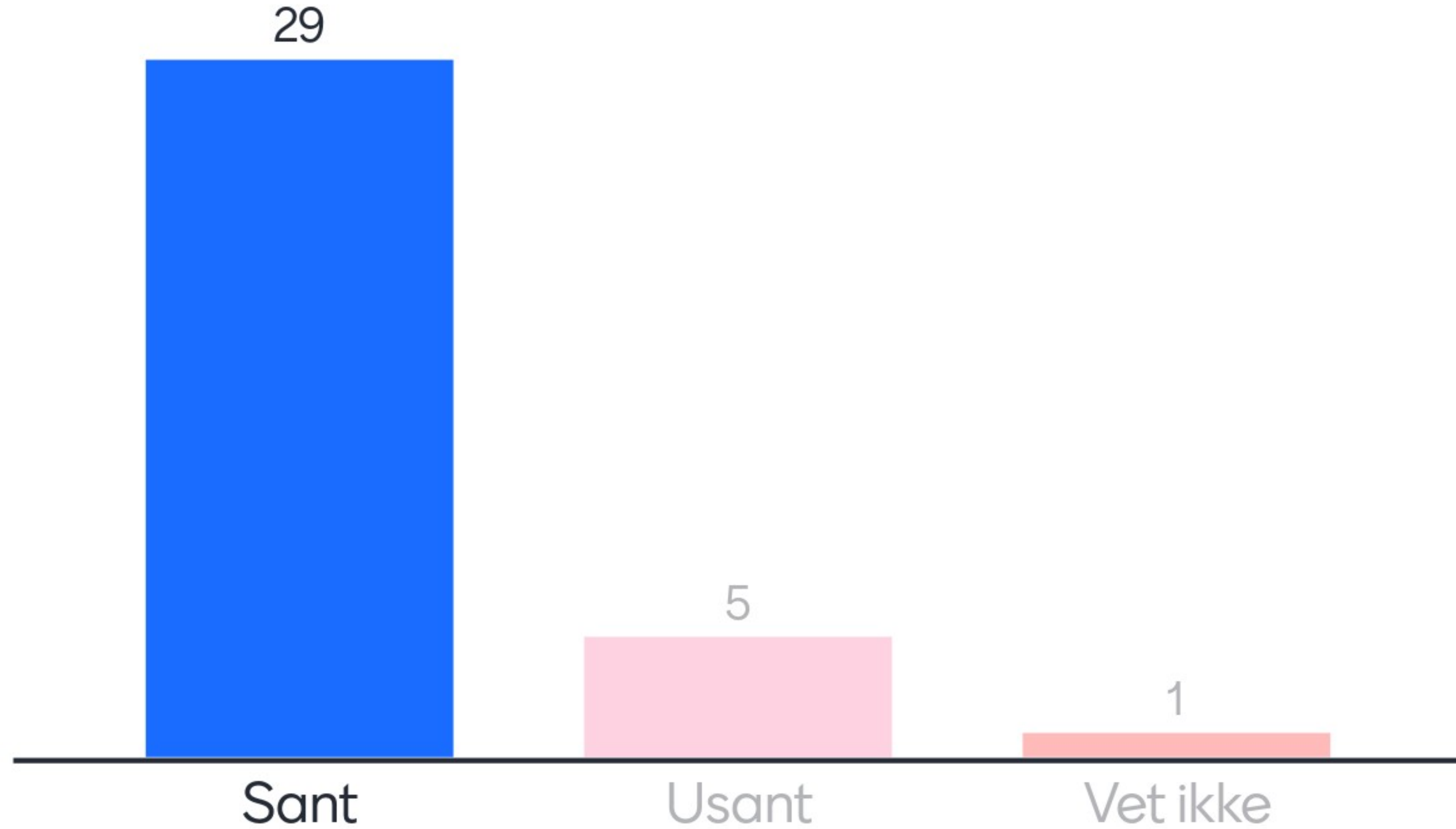
"List" er et interface i Java



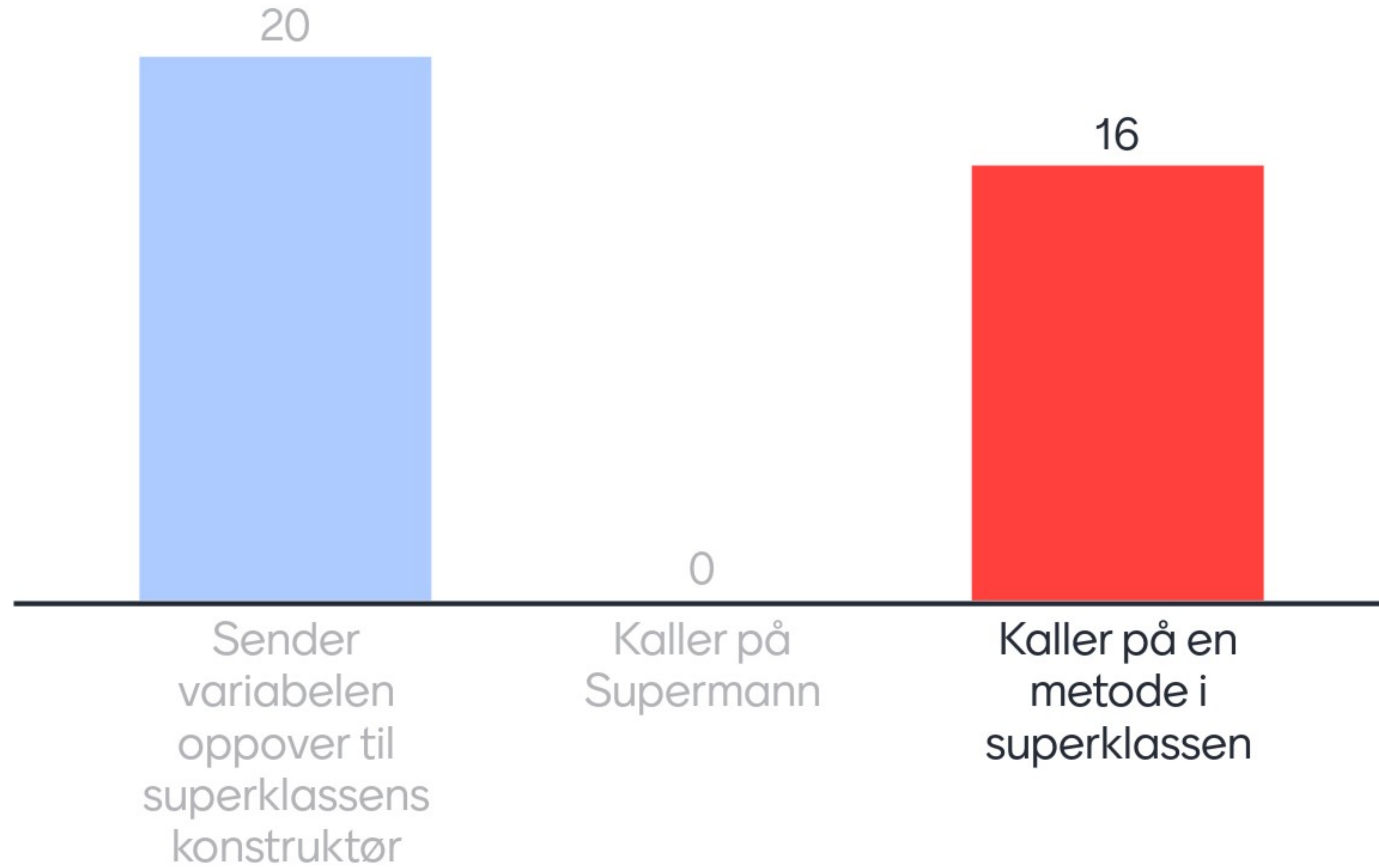
En klasse kan arve fra flere superklasser i Java



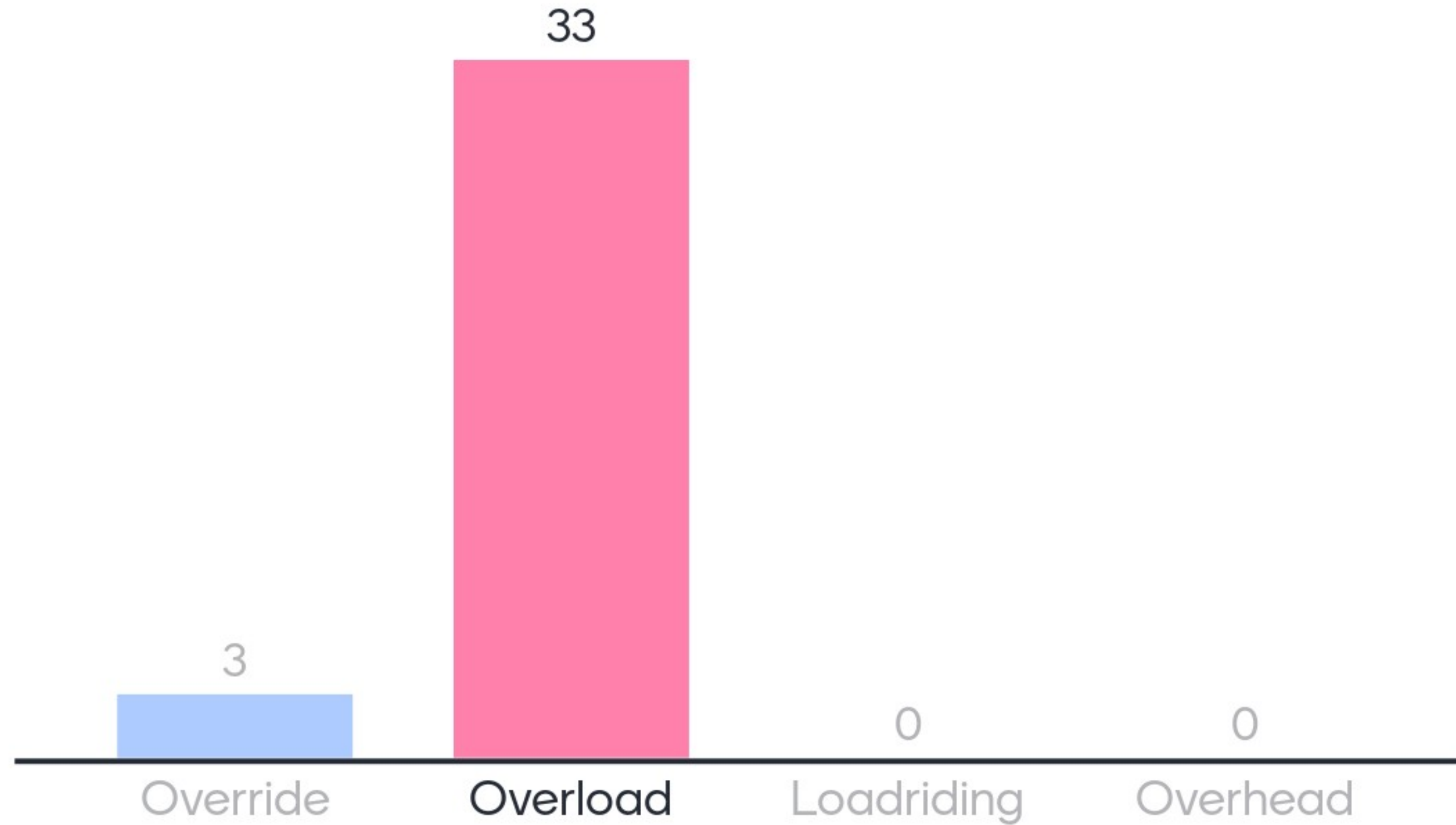
Et interface kan brukes som referansetype



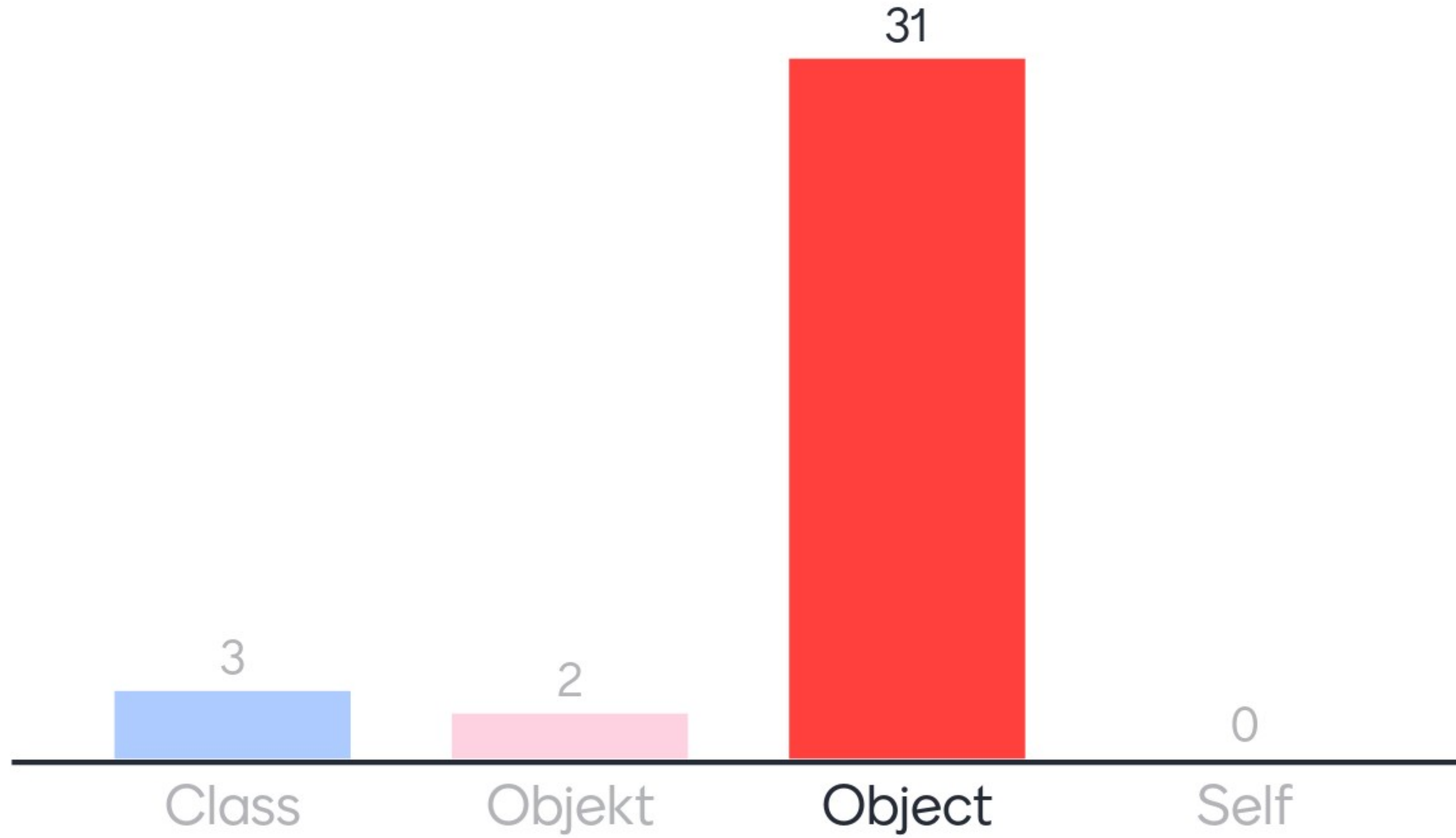
Å skrive "super."



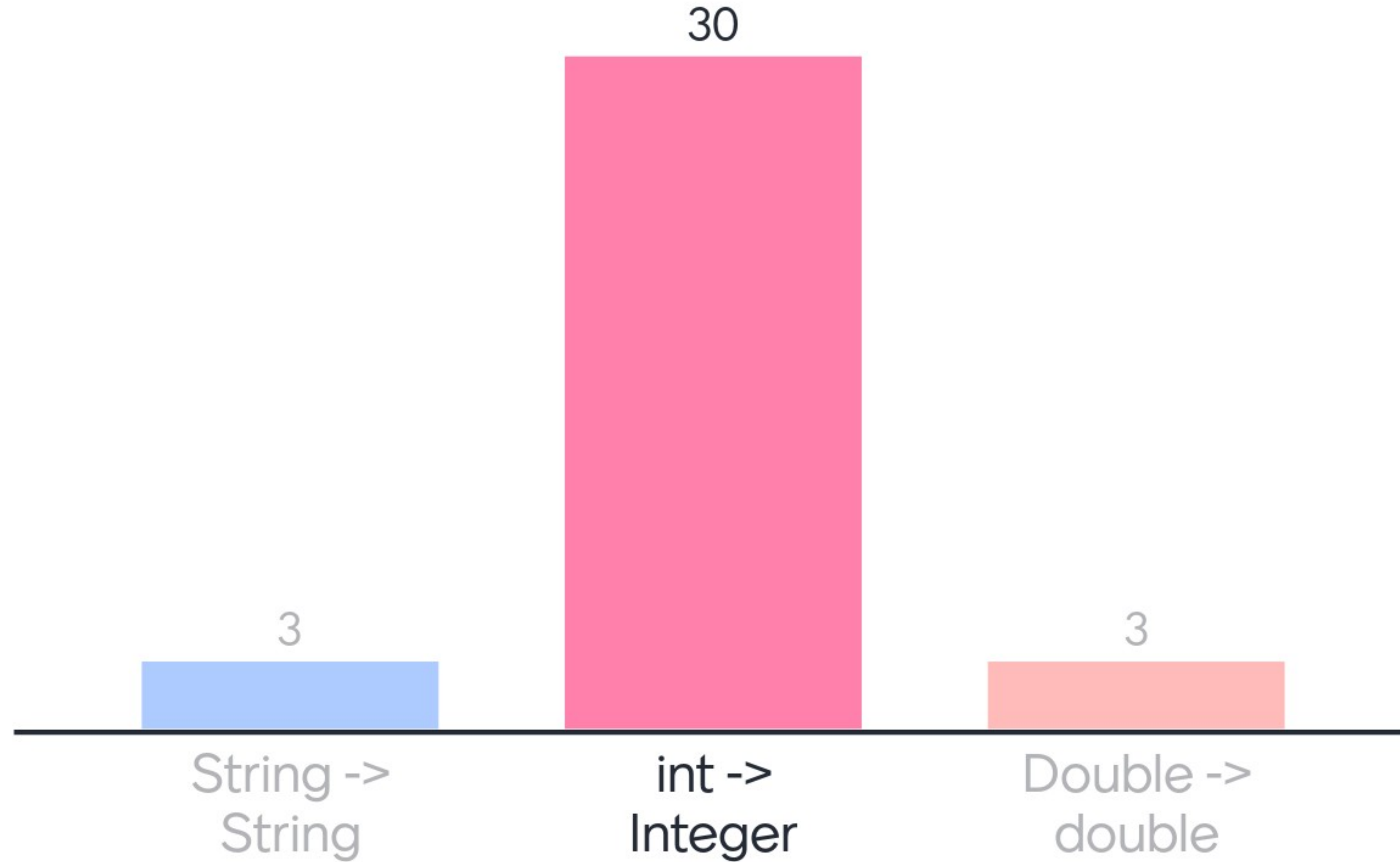
Å definere metoder med like navn men ulike signaturer kalles



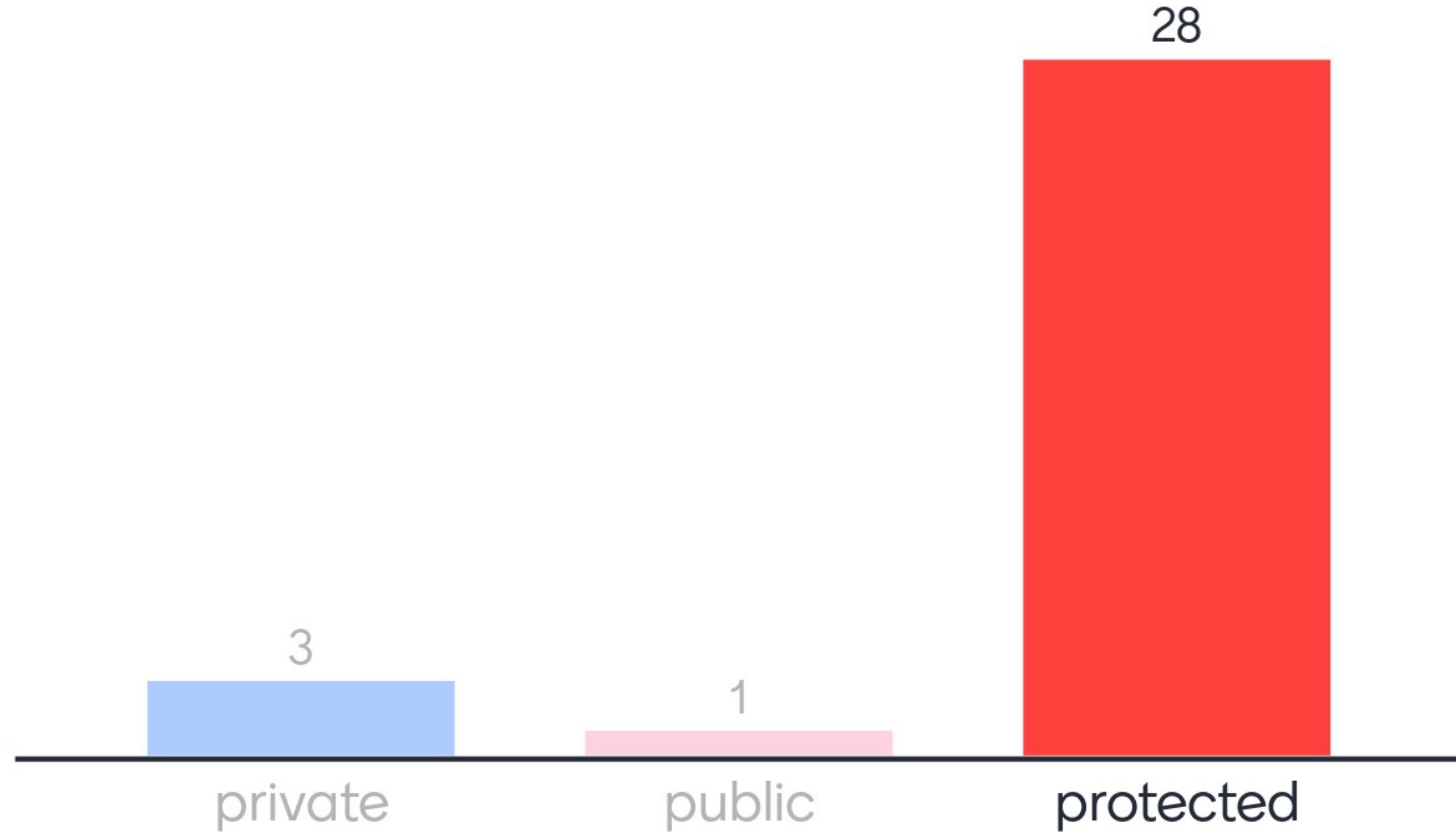
Alle klassers mor...



Et eksempel på "boxing"



Gjør at kun subklasser kan se egenskapen



Forskjellen mellom en abstrakt klasse og et grensesnitt

- Abstrakte klasser kan inneholde fullstendige implementasjoner
- Grensesnitt: kun signaturer
- Ingen innmat i grensesnitt (lite unntak)

Abstrakte klasser på eksamen

- Står sjeldent: "skriv den abstrakte klassen ..."
- Lese mellom linjene
- Eks. "Alle stier er *enten* kjerreveier eller naturstier"
- Nøkkelord som: *enten, kun, det ene eller det andre, skal ikke gå an å lage objekter av*